

**AMENDMENTS TO THE CLAIMS**

**This listing of claims will replace all prior versions and listings of claims in the application:**

**LISTING OF CLAIMS:**

1. (original): A method for receiving a plurality of packets from a network and distributing the packets to a plurality of protocol processors comprising the steps of:

if a received packet is a fragmented packet, determining whether the received packet is a first fragment packet;

if the received packet is the first fragment packet, looking-up a tunnel ID of the received packet and a fragment ID of the received packet, and comparing the result of the looked-up fragment ID with each list of a fragment look-up table into which the results of fragment look-ups for other received packets are entered, to determine if there is a corresponding list;

searching an index indicating one of the protocol processors and corresponding to the tunnel ID of the received packet from a tunnel ID look-up table, and if the list corresponding to the result of the looked-up fragment ID exists in the fragment look-up table, entering the index into the corresponding list of the fragment look-up table; and

attaching the index as a tag to the received packet and transmitting the received packet to the corresponding one of the protocol processors.

2. (original): The method of claim 1, wherein the step of entering the index into the corresponding list of the fragment look-up table, includes newly entering the result of the looked-up fragment ID and the index into the fragment look-up table, if the list corresponding to the result of the looked-up fragment ID does not exist in the fragment look-up table.

3. (original): The method of claim 1, wherein the step of transmitting the received packet includes attaching the index as the tag to a packet that has been previously received and stored in a fragment buffer and transmitting the previously received and stored packet to the corresponding one of the protocol processors, if the received packet is the first fragment and the list corresponding to the result of the looked-up fragment exists in the fragment look-up table.

4. (original): The method of claim 1, wherein if the received packet is not the first fragment, further comprising the steps of:

looking-up the fragment ID of the received packet and comparing the result of the looked-up fragment ID with each list of the fragment look-up table, to determine if there is a corresponding list;

entering the result of the fragment ID looked-up for the received packet into the fragment look-up table, if the list corresponding to the result of the looked-up fragment does not exist in the fragment look-up table; and

storing the received packet in a fragment buffer.

5. (original): The method of claim 4, wherein if the list corresponding to the result of the looked-up fragment ID exists in the fragment look-up table, further comprising the steps of:

determining whether the index corresponding to the result of the tunnel ID look-up exists in the corresponding list; and

attaching the index as the tag to the received packet and transmitting the received packet to the corresponding one of the protocol processors, if the index exists in the corresponding list.

6. (original): The method of claim 5, further comprising the step of storing the received packet in the fragment buffer, if the index does not exist in the corresponding list.

7. (original): An apparatus for distributing a plurality of packets to a plurality of protocol processors comprising:

a receiving unit for receiving the packets from a network;

a fragment look-up table storage unit for storing fragment look-up table into which the result of a fragment looked-up on the received packet is entered;

a fragment look-up device for comparing the result of the fragment looked-up on the received packet with each list of the fragment look-up table, to determine whether the list corresponding to the result exists;

a tunnel ID look-up table storage unit for storing a tunnel ID look-up table having lists of indexes indicating the protocol processors corresponding to the tunnel IDs of the packets, respectively;

a tunnel ID look-up device for searching the index corresponding to the result of the tunnel ID looked-up on the received packet from the tunnel ID look-up table to attach the index as a tag to the received packet; and

a dependant interface for transmitting the packet attached with the index to the corresponding one of the protocol processors.

8. (original): The apparatus of claim 7, wherein if the list corresponding to the result of the looked-up fragment does not exist in the fragment look-up table, the fragment look-up device

newly enters the result of the looked-up fragment and the index into the fragment look-up table, if the received packet is a first fragment, and newly enters the result of the looked-up fragment into the fragment look-up table, if the received packet is not the first fragment.

9. (original): The apparatus of claim 7, further comprising, if the list corresponding to the result of the looked-up fragment and including the index does not exist in the fragment look-up table, a fragment buffer for storing the received packet if the received packet is not the first fragment.

10. (original): The apparatus of claim 9, wherein if the list corresponding to the result of the looked-up fragment and including the index exists in the fragment look-up table, the fragment look-up device attaches the index as the tag to the received packet to transmit the received packet to the corresponding one of the protocol processors.

11. (original): The apparatus of claim 9, wherein in the case of the received packet being the first fragment, the fragment look-up device attaches the index as the tag to each packet being a subsequent fragment following the first fragment and being stored in the fragment buffer to transmit each subsequent fragment packet via the dependant interface to the corresponding one of the protocol processors, if the list conforming to the result of the looked-up fragment exists in the fragment look-up table.

12. (currently amended): The method of claim 1, wherein the other received fragment packets are stored in a fragment buffer, wherein a list is stored in the fragment look-up table for

each of fragmented packets, wherein the fragment look-up table is stored separately from the fragment memory, wherein, if the received fragment packet is determined to be the first fragment packet-is-received, searching the look-up table for a first list that corresponds to the other received fragment packets, wherein the other received fragment packets together with the first fragment packet form a datagram.

13. (currently amended): The method of claim 12, wherein, if ~~when the~~ received fragment packet is determined to be the first fragment packet-is-received, ~~and~~ the first list is found in the look-up table, editing the list to update the index and searching the fragment buffer for the other received fragment packets and transmitting the found other received fragments based on the updated index of the first list without assembling the fragment packets to form the datagram.

14. (currently amended): The method of claim 13, wherein if the received fragment packet is determined to be one of the other fragment packets-is-received, searching the look-up table for the first list, and if the first list is not present, generating the first list comprising source address, destination address and an index and storing the one of the other fragment packets in the fragment buffer.

15. (previously presented): The method of claim 14, wherein the fragment look-up table further comprises a field indicating storage location of respective at least one other fragment packet in the fragment buffer.

16. (previously presented): The method of claim 1, wherein, if the received packet is the first fragment packet, searching an index indicating one of the protocol processors and corresponding to the tunnel ID of the received packet from a tunnel ID look-up table, and if the list corresponding to the result of the looked-up fragment ID exists in the fragment look-up table, updating the index into the corresponding list of the fragment look-up table; and transmitting the other received fragment packets stored in a fragment buffer based on the updated index stored in the corresponding list of the fragment look up table.